

## FINAL PROJECT PROPOSAL

**Project Participants:** Ana Gonzalez

**Title:** ArtConnect: An Art Gallery Management System

**Youtube Link:** <https://youtu.be/ZDoRvuPACNc>

**GitHub Repo:** <https://github.com/MrsDraka/week16-18-final-project.git>

### Executive Summary:

The art gallery API allows any art gallery to keep track of their artworks with the respective artists, and expos. The art gallery would be able to schedule new expos, choose which art pieces will be part of each expos, as well as viewing past, present and future expos.

### Initial Features:

- Entities:
  - artist - 1 artist to many art pieces (1 to many)
  - art\_piece - many art pieces to many expos (many to many)
  - expo - many expos with many art pieces (many to many)
  - art\_piece\_expo - join table
- CRUD Operations:
  - **artist (CRUD)** - POST (create), GET (read), PUT (update), DELETE (delete), GET all, GET by ID
  - **art\_piece (CRUD)** - POST (create), GET (read), PUT (update), DELETE (delete), GET all, GET by ID
  - **expo (CRUD)** - POST (create), GET (read), PUT (update), DELETE (delete), GET all, GET by ID
  - **art\_piece\_expo-join table (C D)** - POST (create), DELETE (delete)
- View a list of all artists
- View an artist and all its data and artworks
- View a list of all artworks
- View the information of a specific artwork, including title, artist, and expos it's been showcased in
- Search expos by date range or location

### Entity Details

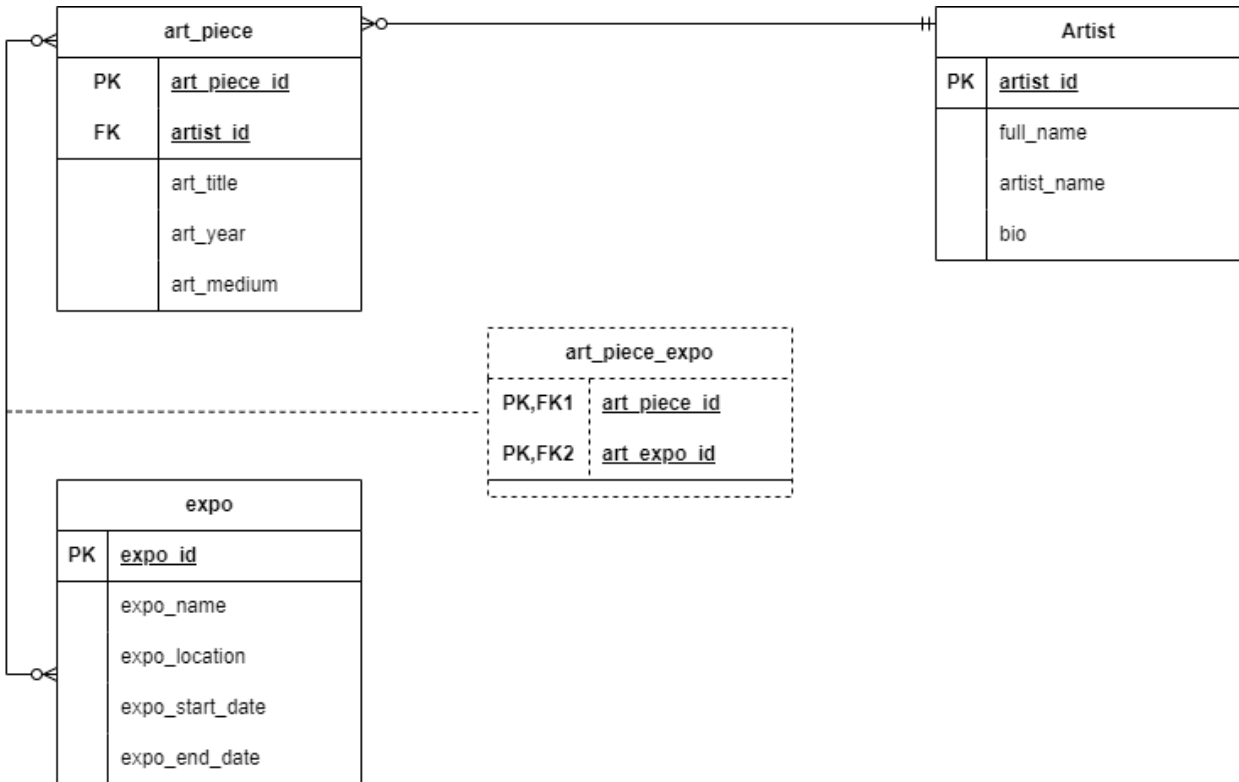
- **Artist table:** Has a unique artist ID, artist name, full name and bio.
- **Art piece table:** Has a unique art piece ID, and it specifies the title, year and medium.
- **Expo table:** Will contain unique expo ID, expo name, location, start date and end date.

### Relationship Information

- The **artist** and **art piece** tables have a **mandatory one to many relationship** since an artist can have many art pieces. The artist is referenced in the **art piece table** by using the artist's id as a foreign key.

- An art piece can be in different expos, and an expo can have several art pieces, hence, the **expo** and **art piece** tables have an **optional many to many relationship** and are referenced using the **art piece expo** join table.

#### Database Entity Relationship Diagram - ERD



#### Description of Endpoint & Features

##### */artists*

Verb	Path Parameters	Query Parameters	Request Body	Response
GET	none	none	none	200 A list of all artists and their info
POST	none	none	full_name, artist_name, bio	201 artist_id, full_name, artist_name, bio

### ***/artists/{artist\_id}***

Verb	Path Parameters	Query Parameters	Request Body	Response
GET	artist_id	none	none	200 Artist info and all art pieces associated with it
PUT	artist_id	none	full_name, artist_name, bio	202 updated artist
DELETE	artist_id	none	none	204 no content

### ***/art\_piece***

Verb	Path Parameters	Query Parameters	Request Body	Response
GET	none	none	none	200 A list of all art pieces and their info
POST	none	none	artist_id, art_title, art_year, art_medium	201 art_piece_id, artist_id, art_title, art_year, art_medium

### ***/art\_piece/{art\_piece\_id}***

Verb	Path Parameters	Query Parameters	Request Body	Response
GET	art_piece_id	none	none	200 All art piece info and the artist associated with it along with the expos it is in, and/or has been in
PUT	art_piece_id	none	artist_id, art_title, art_year, art_medium	202 updated art piece
DELETE	art_piece_id	none	none	204 no content

### ***/expos***

Verb	Path Parameters	Query Parameters	Request Body	Response
GET	none	none	none	200 A list of all art expos and their info
POST	none	none	expo_name, expo_location, expo_start_date, expo_end_date	201 expo_id, expo_name, expo_location, expo_start_date, expo_end_date
GET	none	expo_start_date, expo_end_date	none	200 List of art expos in the given date range

### ***/expos/{expo\_id}***

Verb	Path Parameters	Query Parameters	Request Body	Response
GET	expo_id	none	none	200 All expo info and the art pieces associated with it
PUT	expo_id	none	expo_name, expo_location, expo_start_date, expo_end_date	202 updated expo info
DELETE	expo_id	none	none	204 no content

### ***/expos/{expo\_id}/{art\_piece\_id}***

Verb	Path Parameters	Query Parameters	Request Body	Response
POST	expo_id, art_piece_id	none	none	200 none
DELETE	expo_id, art_piece_id	none	none	204 no content

